

The Relations in Civil Law: A Category Theory Perspective

February 22, 2026

Usayui39^{†1*}, DeepSeek^{‡2*}, ChatGPT^{‡2*}, Claude^{‡2*}, DouBao^{‡2*}

^{†1}Independent Researcher

^{‡2}AI Language Models

*These authors contributed equally to this work

Authors' Note: This paper represents a preliminary theoretical exploration. It was written collaboratively by the first author and multiple AI language models (DeepSeek, ChatGPT, Claude, DouBao). This paper is conceptional and not yet formally verified. Its aim is to provide a unified descriptive language for civil law relations and to clarify possible structural correspondences. The arguments are exploratory and open to further refinement.

Abstract

Formal modelling provides the computational-theoretic foundations necessary for the advancement of computational law and legal artificial intelligence. Taking Chinese civil law as a case study, this paper attempts to develop a structured categorical foundation for the analytical framework of civil law using category theory. Our point of departure is the observation that civil law can, to a large extent, be understood as a closed formal system governing the *relations* between legal subjects — a methodological stance that resonates with the post-structuralist semiotic motif that “there is nothing outside the text (*il n’y a pas de hors-texte*).” We construct a partial category $\mathcal{R}$ whose objects are legal subjects and whose morphisms are civil legal relations; the qualifier “partial” reflects the fact that not every formal composite of two legal relations is itself legally recognised. The three basic operations on civil legal relations — creation, extinction, and modification — are characterised, respectively, by an extension functor, by direct removal of a morphism (with discussion of its non-functorial character), and by natural transformations between functors (with a careful disambiguation from the notation used for the monad unit). The concurrence of rights over the same object is expressed using the diagrammatic language of limits and colimits, with an explicit correction of the common mis-identification of the

coequalizer with proportional distribution. For the nesting of relations (e.g. a security interest accessory to a principal claim), functor composition is used. We then argue that the totality of civil law mechanisms can be viewed as a *monad* template (T, η_T, μ) on $\mathcal{R}$: the monad unit corresponds to the legal recognition of “bare social facts,” and monad multiplication corresponds to the sequential application of legal acts. The paper concludes by replacing the incorrect functor-category formulation $[\mathcal{T}, \mathcal{R}]$ with the correct model of a pseudofunctor $\Phi : \mathcal{T} \rightarrow \mathbf{Cat}$, under which the system at each instant is a *category* $\mathcal{R}_t$ rather than a single object of $\mathcal{R}$, and discusses how statutes of limitations and peremptory periods arise within this framework.

Keywords: civil law, category theory, partial category, relational theory, formal modelling, computational law

1 Introduction

A direct motivation for discussing formal analysis in legal scholarship is the following: as computational law and legal artificial intelligence develop, researchers need to convert legal concepts, legal relations, and institutional operations into representable, operable, and inferable structures, in order to support tasks such as knowledge representation, automated reasoning, retrieval, and decision assistance. Formal modelling is not an end in itself but a foundational step enabling computational modelling; and before entering into specific formal representations, it is usually necessary to achieve a degree of ontological clarification about the object of modelling: precisely which “elements” are to be identified in the model, and in what manner these elements are related to one another.

Work [1] proposes that social facts can be decomposed into six interrelated elements: entities, attributes, states, processes, events, and relations. In a certain sense, the subject matter studied by legal scholarship and disciplines such as sociology bears structural similarities: both revolve around “factual states” that carry normative or social significance. Although legal scholarship is not equivalent to empirical social science — its core concern lies in normative structures and institutional arrangements — when undertaking formal modelling we still face a basic question: into what fundamental constituent elements can the “object world” that law addresses be decomposed, and how are these elements organised?

Concretely, the social world addressed by civil law can be analysed along these six dimensions. *Entities* are the bearers of civil life, including natural persons, legal persons, unincorporated organisations, and things. *Attributes* are properties intrinsic to an entity and independent of relations, such as civil capacity or the natural properties of a thing. *States* are the legal conditions of an entity that persist through time, such as being an adult, being missing, or being bankrupt. *Processes* are the directed changes of the legal-relation system, manifested as the creation, transfer, and extinction of rights. *Events* are external facts that trigger legal effects, including birth, death, natural disasters, and declarations of intent. *Relations* are structured legal connections between entities, and they constitute the core object of civil law.

Among these elements, the legal relation (*Rechtsverhältnis*) occupies a central position. This contrasts sharply with criminal law, which centres on conduct (*Handlung*), and administrative law, which centres on the relation of public authority. The legal relation functions as a hub: it links entities together, takes events as catalysts, undergoes change through processes, and remains constrained throughout by states.

The correspondences in Table 1 form the starting point of our modelling. A special note is due regarding the status of “objects of rights”: in the modelling strategy adopted here (see Section 2.1), things and other objects of rights do not appear directly as objects of the cate-

Table 1: Correspondence between ontological elements and civil law concepts

Ontological Category	Corresponding Civil Law Concept
Entity	Legal subject / object of rights
Attribute	Civil capacity; properties of a thing
State	Legal status (adult, missing, owner, etc.)
Process	Creation, modification, extinction of legal relations
Event	Legal fact (natural event / juridical act)
Relation	Civil legal relation

gory, but are expressed indirectly through *in rem* relations. This reflects our methodological commitment to “relation-centrism.”

Why choose category theory as the tool for formalising civil law? The choice is grounded in structural characteristics of civil law itself. First, the essence of category theory is “morphisms take priority over objects” — objects are defined and characterised through their morphisms to and from other objects. This aligns with the relational ontology of civil law: a legal subject’s significance is precisely conferred through the network of legal relations in which it participates. Second, civil law is a closed normative system: the validity of juridical acts derives from the system’s internal recognition, and whether acts can be combined is subject to explicit structural rules. Category theory provides a precise language for studying such “combinatorics.” Third, civil law exhibits obvious level distinctions, e.g. between a “bare social fact” and a “legal fact,” or between a “substantive right” and “actionability.” Functors as “structure-preserving maps” are the natural tool for expressing such inter-level relations. Finally, category theory possesses a rich vocabulary of higher-order structural concepts — monads, natural transformations, limits, colimits — which offer potential descriptive tools for complex phenomena.

It must be emphasised that the goal of this paper is to construct a *formal framework* rather than a *mathematical model*. A formal framework provides a descriptive language and a conceptual tool, aiming to clarify conceptual structure and reveal hidden assumptions, without necessarily achieving mathematical computability or predictability. This methodological stance will be maintained throughout.

2 Construction of the Partial Category $\mathcal{R}$ of Legal Relations

2.1 Two Construction Paths

In the first step of “categorifying” civil law, we face a fundamental choice: what should be objects, and what should be morphisms? Table 2 contrasts two possible paths.

This paper takes Reading A as its main axis, and compensates for its temporal deficiency in

Table 2: Comparison of two construction paths for the category of legal relations

Dimension	Reading A: Subjects as objects, relations as morphisms	Reading B: System states as objects, juridical acts/events as morphisms
Object definition	Legal subjects (natural persons, legal persons, etc.)	Global snapshot of the relation system at a given moment
Morphism definition	Civil legal relations between subjects	Juridical acts or events transforming state S_1 into S_2
Identity morphism	A technical placeholder; see discussion below	The “null event” transforming a state into itself
Advantages	Aligned with relational ontology; facilitates characterisation of internal structure	Facilitates handling temporality and procedurality; compatible with litigation procedure
Disadvantages	Temporality must be introduced externally; composition only partial	Difficult to express directly the internal constitution of a relation
Handling concurrence	Multiple morphisms converging on the same object/thing	Multiple transformation paths on the same state

102 Section 6.

Definition 2.1 — Partial Category $\mathcal{R}$ (Reading A)

We define $\mathcal{R}$ as a *partial category*^a as follows:

$$\text{Ob}(\mathcal{R}) = \text{the set of all civil legal subjects } (A, B, C, \dots)$$

$$\text{Hom}_{\mathcal{R}}(A, B) = \text{the set of legally valid civil relations directed from } A \text{ to } B.$$

Directionality and bilateral obligations. A civil legal relation between two parties is, in general, bilateral: a creditor holds a right to claim, and a debtor bears a correlative duty to perform. Directing a morphism $r \in \text{Hom}(A, B)$ from A (holder of the primary entitlement) to B (bearer of the primary duty) is therefore a modelling choice that privileges one side of the relation. Where the bilateral structure is important — e.g. when both parties bear obligations — a relation may be represented by a pair of morphisms $(r_1 : A \rightarrow B, r_2 : B \rightarrow A)$ or, where symmetry holds, by a single undirected edge. The paper adopts the directed convention for conciseness, with the understanding that this simplification is acknowledged.

Composition. For $f \in \text{Hom}(A, B)$ and $g \in \text{Hom}(B, C)$, the composite $g \circ_{\text{Law}} f$ is *defined* if and only if law recognises a directly valid A -to- C relation derived from f and g ; when defined, $g \circ_{\text{Law}} f \in \text{Hom}(A, C)$ is that derived relation. We write $\circ_{\text{Law}}$ throughout to distinguish *lawful composition* from the always-defined formal composite; the latter may exist as a set-theoretic object without carrying legal validity.

Table 3: Basic types of civil legal relations and their categorical characteristics

Morphism Family	Structural Feature	Effectiveness Feature	Civil Law Example
Personal relations	$A \leftrightarrow B$ (representable by a symmetric pair of morphisms)	Related to status; often continuous	Spousal relation; parent-child relation
<i>In rem</i> relations	$A \curvearrowright X$ (dominion of person over thing; erga omnes effect mediated by thing X) ¹	Absolute; erga omnes; exclusive	Ownership; usufructuary rights
Obligational relations	$A \rightarrow B$ (primary entitlement-holder to primary duty-bearer)	Relative; inter partes; claimatory	Contractual claims; tort damages claims

Identity morphisms. Each object A is equipped with a placeholder identity morphism $\text{id}_A \in \text{Hom}(A, A)$ satisfying $f \circ_{\text{Law}} \text{id}_A = f$ and $\text{id}_B \circ_{\text{Law}} f = f$ wherever those composites are defined. We stress that id_A is a *mathematical necessity* for the (partial) category axioms; it does not correspond to any substantive civil law institution. The earlier suggestion of equating id_A with “legal capacity from birth to death” (Art. 13 Civil Code) is an error of category: legal capacity is an attribute of a subject, not a self-directed legal relation. We therefore offer no legal interpretation for id_A beyond its formal role.

^aA partial category is a directed graph (objects and morphisms) equipped with a partially defined associative composition and identity morphisms, satisfying unit laws wherever composition is defined. See Cockett and Lack, *Restriction categories I: categories of partial maps*, Theoretical Computer Science **270** (2002), 223–259, for a thorough treatment of partial composition structures.

Remark 2.1. The partial-category structure of $\mathcal{R}$ deserves emphasis. The standard category axioms require composition to be everywhere defined on composable pairs and to be associative. In $\mathcal{R}$, composition is only partially defined (it holds wherever law recognises the derived relation), but wherever it is defined it is required to be associative. The unit laws $f \circ_{\text{Law}} \text{id}_A = f$ and $\text{id}_B \circ_{\text{Law}} f = f$ must hold for all $f \in \text{Hom}(A, B)$. This makes $\mathcal{R}$ not a category in the standard sense, but it is a well-studied algebraic structure. Throughout the paper, whenever we invoke categorical machinery (functors, natural transformations, limits, monads), the reader should understand these as applied to the partial setting, with appropriate caveats about existence and uniqueness.

2.2 Three Basic Types of Morphism

According to established civil law doctrine, civil legal relations can be classified into three basic types according to their effectiveness characteristics.

117 The classification is not absolute; hybrid relations exist (e.g. a lease combines obligational and
 118 *in rem* features). A morphism may belong to multiple sub-families simultaneously, provided
 119 the corresponding composition rules are explicitly defined.

120 2.3 Lawful Composition: Examples and Cautions

121 The partial composition $\circ_{\text{Law}}$ is the structural heart of $\mathcal{R}$. We discuss two legally significant
 122 cases that were treated imprecisely in earlier versions of this framework.

Proposition 2.1 — Lawful Composition: Subrogation

Consider subrogation (Art. 535 Civil Code). Creditor A holds a claim $f : A \rightarrow B$ against debtor B ; B holds a claim $g : B \rightarrow C$ against sub-debtor C . When the statutory conditions are met — B 's claim against C is due, B has failed to exercise it, and A 's claim against B is thereby endangered — law permits A to exercise B 's claim against C in A 's own name, for the benefit of B 's estate.

123

124 **Remark 2.2** (Subrogation is not a simple composite). Formally writing $g \circ_{\text{Law}} f$ as a single
 125 morphism $A \rightarrow C$ risks misrepresenting the institution. Under Art. 535, what A asserts against
 126 C is B 's claim g , not a novel direct claim of A against C . The effect of a successful subrogation
 127 action is that B 's estate is increased (by C 's payment), from which A then satisfies f . There
 128 are thus two distinct legal steps: (i) A exercises $g : B \rightarrow C$ on behalf of B 's estate; (ii) A
 129 satisfies $f : A \rightarrow B$ from the recovered assets. The composite $g \circ_{\text{Law}} f$, if used at all, should
 130 therefore be understood as a *structural abbreviation* for this two-step process rather than a
 131 single direct entitlement of A against C . Whether a genuine derived morphism $A \rightarrow C$ exists
 132 in $\mathcal{R}$ is itself a legal question that the categorical notation cannot pre-answer. The composite
 133 is conditional, partially defined, and procedurally mediated — all consistent with the partial-
 134 category structure, but the legal substance is more complex than the notation suggests.

135 **Remark 2.3** (Assignment of claims is not a composite). Assignment of claims (Arts. 545–550
 136 Civil Code) is frequently cited as a paradigm of morphism composition: after A assigns claim
 137 $f : A \rightarrow B$ to A' , the new creditor A' holds a claim against B . It is tempting to write this as a
 138 composite $f \circ (\text{assignment} : A' \rightarrow A)$.

139 This is incorrect on two counts. First, the “assignment” from A to A' is not a morphism of $\mathcal{R}$
 140 in the sense of a civil legal relation from A' to A ; rather, it is a *transfer event* that changes the
 141 holder of the right. Second, the substance of what A' holds is the *same* claim (with the same
 142 content, defences, and accessory rights) as f — it is not a new derived claim. Assignment is
 143 therefore better modelled as a *substitution of subject*: the relation f is re-labelled from $A \rightarrow B$
 144 to $A' \rightarrow B$. Section 3 below will characterise this as a natural transformation, which is the
 145 appropriate categorical tool for subject substitution.

Table 4: Different pathways for relation creation and their modelling features

Creation Pathway	Modelling Feature	Civil Law Example
Juridical act	Parties actively introduce a relation through a declaration of intent; the “source” of the new morphism is recorded	Contract formation; unilateral promise
Legal event	Independent of the parties’ intent; law directly attaches a relation-generating effect to the event	Birth (generates parental duty); death (opens succession)
Quasi-juridical act	A certain expression, notice, or attitude triggers relation-generation under applicable rules	Notice of demand (); ratification (); acknowledgment ()
Wrongful act	Law forcibly inserts a relation between parties to correct an unlawful state	Tortious obligation; unjust enrichment obligation

146 3 Dynamics of Relations: Creation, Extinction, and Modification

147 Civil legal relations undergo dynamic processes of creation, extinction, and modification. This
 148 section explores how to characterise these processes categorically.

149 3.1 Creation: Extension of the Category

150 Creating a new relation corresponds to adding a new morphism (and possibly a new object) to
 151 $\mathcal{R}$. Let $\mathcal{R}$ denote the system before the act and $\mathcal{R}'$ the system after; there exists an extension:

$$\iota : \mathcal{R} \hookrightarrow \mathcal{R}'$$

152 preserving all existing objects, morphisms, and defined composites. In most everyday situa-
 153 tions the object set is unchanged ($\text{Ob}(\mathcal{R}) = \text{Ob}(\mathcal{R}')$) and only new morphisms are added.

154 All pathways in Table 4 share the structure of an extension ι , but differ in their “generative
 155 mechanism.” These mechanisms involve the transition from “social fact” to “legal relation,”
 156 which is the entry problem addressed by the monad model in Section 5.

157 3.2 Extinction: Removal of a Morphism

158 Extinction of a relation means that a particular morphism $r \in \text{Hom}_{\mathcal{R}}(A, B)$ loses its legal force.
 159 Extinction is modelled as *removal* of the morphism from the category, not as “collapsing to
 160 the identity” (which would leave a ghost of the relation and distort the unit laws).

Table 5: Typical mechanisms of relation extinction

Extinction Type	Legal Mechanism	Modelling Treatment
Performance	Full realisation in accordance with the content of the obligation	Remove r ; cascade to all accessory morphisms depending on r
Avoidance/rescission	Deprives the juridical act of effect from inception or for the future	Remove r ; optionally annotate a record for reliance-interest protection
Merger ()	Creditor and debtor become the same subject	When $A = B$, $r \in \text{Hom}(A, A)$ is removed. Note: removal of a self-loop does not disrupt composition of other morphisms through A
Limitation period expired	Upon expiry, the actionability of the claim is diminished but the substantive relation is not extinguished	The substantive relation r is retained; the “actionability layer” $\tilde{r}$ is removed (see Section 6)
Judicial intervention	A court judgment causes extinction	External functor $J : \mathcal{R} \xrightarrow{\text{judgment}} \mathcal{R}'$

161 Let $\mathcal{R}$ be the category before extinction. The new category $\mathcal{R}'$ obtained after extinction of r
 162 satisfies:

$$\text{Ob}(\mathcal{R}') = \text{Ob}(\mathcal{R})$$

$$\text{Hom}_{\mathcal{R}'}(X, Y) = \begin{cases} \text{Hom}_{\mathcal{R}}(X, Y) \setminus \{r\} & \text{if } (X, Y) = (A, B) \\ \text{Hom}_{\mathcal{R}}(X, Y) & \text{otherwise} \end{cases}$$

163 **Remark 3.1** (Cascading effects of extinction). Removing $r : A \rightarrow B$ may invalidate compos-
 164 ites that passed through r . Any morphism of the form $r \circ_{\text{Law}} f$ or $g \circ_{\text{Law}} r$ that was previously
 165 defined in $\mathcal{R}$ will no longer be defined in $\mathcal{R}'$, because its constituent r no longer exists. Fail-
 166 ure to address this cascading effect would leave $\mathcal{R}'$ with partially orphaned composition data.
 167 In practice, this cascading corresponds to the legal principle that accessory rights and obli-
 168 gations (over a principal debt, for instance) are extinguished when the principal relation is
 169 extinguished (Art. 407 Civil Code). The model should propagate extinction to all accessory
 170 morphisms depending on r .

171 Importantly, there is no natural functor from $\mathcal{R}$ to $\mathcal{R}'$ in general: extinction is a *discontinuous*
 172 structural change. This distinguishes extinction from creation (ι is a functor) and from modi-
 173 fication (expressible by a natural transformation; see below). The asymmetry is not a defect of
 174 the model but reflects a genuine structural difference among the three operations in civil law.

175 The distinction in the fourth row — between the substantive relation r (which survives the
 176 limitation period) and the actionability layer $\tilde{r}$ (which does not) — reflects a layered structure
 177 of legal morphisms. This point is developed in Section 6.

178 3.3 Modification: Natural Transformations as Coherence Constraints

179 Modification adjusts the content or the subjects of a relation while preserving its continuity,
 180 and requires that the surrounding relational network remain coherent. Natural transformations
 181 provide a precise language for this coherence requirement.

182 **Notation alert.** In Section 5 below, the symbol η_T is reserved for the *monad unit*. In the
 183 present section, we use the symbol α (with subscripts) for natural transformations expressing
 184 modification, to avoid any confusion.

185 Let $F, G : \mathcal{R} \rightarrow \mathcal{R}$ be two functors representing the “relational interpretation frameworks”
 186 before and after modification respectively. A modification is expressed by a natural transfor-
 187 mation $\alpha : F \Rightarrow G$ whose components $\alpha_A : F(A) \rightarrow G(A)$ satisfy, for every $r \in \text{Hom}_{\mathcal{R}}(A, B)$
 188 for which both $F(r)$ and $G(r)$ are defined, the commutative diagram:

$$\begin{array}{ccc} F(A) & \xrightarrow{\alpha_A} & G(A) \\ F(r) \downarrow & & \downarrow G(r) \\ F(B) & \xrightarrow{\alpha_B} & G(B) \end{array}$$

189 **The case of content modification** (e.g. adjusting the amount of a contractual debt). If only
 190 the content of a relation changes but the subjects remain, F and G may both be taken as the
 191 identity functor $\text{Id}_{\mathcal{R}}$. A natural transformation $\alpha : \text{Id} \Rightarrow \text{Id}$ then has components $\alpha_A : A \rightarrow A$,
 192 i.e. morphisms from A to itself in $\mathcal{R}$. This means content modification is modelled as a special
 193 “self-relation” of each subject. The commutativity condition becomes:

$$\begin{array}{ccc} A & \xrightarrow{\alpha_A} & A \\ r \downarrow & & \downarrow r \\ B & \xrightarrow{\alpha_B} & B \end{array} \quad \text{i.e.,} \quad r \circ_{\text{Law}} \alpha_A = \alpha_B \circ_{\text{Law}} r$$

194 for all $r : A \rightarrow B$ for which both composites are defined.

Proposition 3.1 — Coherence Condition for Modification

A content modification $\alpha : \text{Id} \Rightarrow \text{Id}$ must commute (in the sense of the diagram above) with every existing legal relation r for which both composites are defined in the partial category. This expresses the requirement that a modification not disrupt the coherence of the existing

195

Table 6: Categorical characterisation of types of modification

Modification Type	Categorical Expression	Civil Law Example
Content modification	$\alpha : \text{Id} \Rightarrow \text{Id}$; components $\alpha_A : A \rightarrow A$	Contractual modification of price or quantity; adjustment for changed circumstances ()
Subject substitution	$\alpha : \text{Id} \Rightarrow G$ where G is a subject-replacement functor; captures same content, new holder	Assignment of claim: $f : A \rightarrow B$ becomes $f' : A' \rightarrow B$ with same defences and accessory rights
Object substitution	Morphism label adjusted; subjects unchanged	Substitution of performance (): delivery of X in lieu of Y

relational network.

196

Remark 3.2. Two important qualifications attach to Proposition 3.1.

197

First, the partial-category setting substantially weakens the universality of the condition. In a full category, commutativity with “all” r is a stringent global constraint. In $\mathcal{R}$, it applies only to those r for which $r \circ_{\text{Law}} \alpha_A$ and $\alpha_B \circ_{\text{Law}} r$ are both defined. Many relations r that are unrelated to the modification will fail to compose with α_A or α_B at all, and hence impose no constraint. The condition is therefore far less restrictive than it appears.

198

199

200

201

202

Second, the proposition captures a structural ideal: that a modification propagates consistently through all related chains. In practice, legal systems achieve this ideal procedurally — e.g. by requiring notification to the debtor upon assignment (Art. 546 Civil Code) — rather than by simultaneous global satisfaction of all commutativity conditions.

203

204

205

206

The case of subject substitution (e.g. assignment of claims, Remark 2.3). Here F and G are not identity functors. Define $F = \text{Id}_{\mathcal{R}}$ and G to be the functor that replaces subject A by a new subject A' while fixing all other objects. A natural transformation $\alpha : \text{Id} \Rightarrow G$ then has components $\alpha_A : A \rightarrow G(A) = A'$ (not a morphism within $\mathcal{R}$ unless A and A' are already objects of $\mathcal{R}$) and $\alpha_B = \text{id}_B$ for $B \neq A$. The commutativity condition $G(r) \circ \alpha_A = \alpha_B \circ r$ becomes $r' : A' \rightarrow B$ equals the image of $r : A \rightarrow B$ under the subject substitution. This correctly captures the fact that the assigned claim $r' = f' : A' \rightarrow B$ has the same content as $f : A \rightarrow B$ — it is the same obligation, with a new holder.

207

208

209

210

211

212

213

214

215 4 Concurrency: Structural Expression via Limits and Colimits

216 4.1 Precise Framing of the Concurrency Problem

217 What conflicts in concurrency is not “relations themselves” but the concrete rights, obliga-
 218 tions, or claims derived from them. When different relations directed at the same object or
 219 performance cannot all be simultaneously satisfied, a selection or priority problem arises.

220 In $\mathcal{R}$, concurrency manifests as multiple morphisms $f_i : A \rightarrow B$ (multiple claims of A against B)
 221 or $g_j : A \rightarrow X$ (in rem and obligational claims on the same object X) converging on a “conflict
 222 point.”

223 4.2 The Language of Limits and Colimits

224 The mathematical essence of concurrency is a structural convergence problem: given multiple
 225 morphisms with a common source or target, the system must determine which is preferentially
 226 satisfied. Category theory provides precise language for such convergent structures through
 227 *limits* and *colimits*.

228 Consider the simplest case: two parallel morphisms $f, g : A \rightarrow B$ forming the diagram:

$$A \begin{array}{c} \xrightarrow{f} \\ \xrightarrow{g} \end{array} B$$

229 The **equalizer** of this diagram is a morphism $e : E \rightarrow A$ satisfying $f \circ e = g \circ e$, universal with
 230 this property. It identifies the “largest sub-domain” of A on which f and g agree — i.e. the
 231 conditions under which both claims yield identical results. It does *not* represent the common
 232 basis of the claims in any substantive legal sense.

233 The **coequalizer** of this diagram is a morphism $q : B \rightarrow Q$ satisfying $q \circ f = q \circ g$, universal
 234 with this property. It forms a quotient of B that identifies the image of f with the image of g ,
 235 collapsing the distinction between the two claims’ effects. This is a structural tool for “merg-
 236 ing” two relations into one, not a model of proportional distribution. The earlier identification
 237 of the coequalizer with “pro-rata distribution” is incorrect and is withdrawn: pro-rata distribu-
 238 tion corresponds to a specific legal rule imposed on top of the categorical structure, not to the
 239 coequalizer itself.

Definition 4.1 — Categorical Expression of Concurrency

Let $\mathcal{D}$ be a diagram of morphisms in $\mathcal{R}$ representing multiple legal relations converging on a particular object or subject. The **limit** of $\mathcal{D}$, when it exists, is an object L with morphisms to each node of $\mathcal{D}$, universal with the property that all the diagram’s commutativity conditions

240

Table 7: Exception rules giving priority of obligational rights over *in rem* rights

Exception Rule	Effect	Legal Basis
Sale does not break lease ()	The tenant's right to use and receive profits continues to bind the transferee within the statutory period	Art. 725 Civil Code
Advance notice registration ()	The advance-notice registration gives a contractual claim the effect of opposing subsequent dispositions of the property right	Art. 221 Civil Code
Construction project payment priority ()	Certain construction claims are satisfied in priority over mortgage rights	Art. 807 Civil Code

are satisfied. Intuitively it characterises the “most constrained consistent position” implied by all the convergent relations jointly. The **colimit** of $\mathcal{D}$, when it exists, is an object C with morphisms from each node, universal for the same commutativity data; it characterises the “minimal structure into which all the convergent relations can be embedded consistently.”

The legal *resolution* of concurrence (priority ordering, proportional satisfaction, exclusive assignment) is a further choice made by positive law *after* identifying the relevant limit or colimit structure. Category theory supplies the structural form of the conflict; positive law supplies the resolution rule. Neither replaces the other.

241

Remark 4.1 (Non-existence of limits in $\mathcal{R}$). There is no guarantee that a given diagram in $\mathcal{R}$ has a limit or colimit. Category theory only ensures their existence in categories with appropriate completeness properties (e.g. small-complete categories). $\mathcal{R}$ as defined has no reason to possess these properties. The fact that law *must* resolve every genuine concurrence (courts cannot decline to adjudicate on grounds of categorical incompleteness) shows that the legal resolution is supplied from outside the categorical structure. This “external supplement” — which we call the *judicial completeness operator* J — is discussed in Section 7.

4.3 Property–Obligation Concurrence

The concurrence of *in rem* and obligational relations manifests categorically as an *in rem* relation $f : A \curvearrowright X$ and an obligational relation $g : A \rightarrow B$ converging on thing X . Table 7 lists exception rules where obligational rights prevail.

In each case the categorical structure is similar — two morphisms converge on the same object — but positive law selects which takes priority. The categorical framework locates the structural form of the conflict; the legal content of each exception rule is irreducibly substantive.

Table 8: Priority rules for claim distribution in insolvency

Context	Priority Rule	Legal Basis
Insolvency distribution	Claims satisfied in statutory order: insolvency costs → employee claims → taxes → general claims	Art. 113 Enterprise Insolvency Law
Separate satisfaction right ()	Claims secured by property satisfied from secured asset in priority	Art. 109 Enterprise Insolvency Law
Right of set-off ()	Mutually owed obligations may be set off in insolvency	Art. 40 Enterprise Insolvency Law

256 4.4 Obligation–Obligation Concurrence

257 Multiple claims $f_i : A_i \rightarrow B$ against the same debtor B form a concurrent diagram. Table 8
258 shows the typical treatment in insolvency proceedings.

259 From a categorical viewpoint, the insolvency priority rules correspond to a “selection function”
260 on the set of concurrent morphisms, imposing an ordering on which morphisms are satisfied
261 first when resources are limited. This selection function is supplied entirely by positive law; it
262 is not derivable from the categorical structure alone.

263 5 The Monad as a Structural Template

264 This section proposes a more exploratory structural reading: the operation of civil law as a
265 whole exhibits features that are structurally analogous to a *monad* on $\mathcal{R}$. We use the word
266 “template” deliberately: the claim is not that a specific monad has been constructed and veri-
267 fied, but that the monad’s three components (T, η_T, μ) provide a useful conceptual scaffold.

268 5.1 From “Social Fact” to “Legal Relation”

269 The starting point of civil law’s operation is the “social fact” — contract signings, tortious
270 acts, natural events, etc. These facts are not themselves legal relations; they acquire legal force
271 only through the law’s act of “recognition.” This recognition process can be abstracted as a
272 functor $L : \text{Fact} \rightarrow \mathcal{R}$ from a “social-fact category” Fact (whose objects are social states and
273 whose morphisms are social processes) to $\mathcal{R}$. For conciseness, we internalise this recognition
274 by working with an endofunctor $T : \mathcal{R} \rightarrow \mathcal{R}$ that models the normative filtering of relational
275 data.

Definition 5.1 — Civil Law Monad Template

The civil law monad template on $\mathcal{R}$ consists of (T, η_T, μ) :

- $T : \mathcal{R} \rightarrow \mathcal{R}$ is an endofunctor. Intuitively, $T(A)$ is the “legally recognised version” of subject A and $T(r)$ is the legally recognised image of relation r after being filtered through civil law’s normative mechanism. The precise construction of T is left open at this stage; we treat it as a programmatic object pending further development.
- $\eta_T : \text{Id}_{\mathcal{R}} \Rightarrow T$ is the **unit** natural transformation, with components $(\eta_T)_A : A \rightarrow T(A)$. It corresponds to the legal fact recognition mechanism: the “lifting” of a bare social state into the space of legally evaluated relations. **Note:** we write η_T (not η) throughout this section to distinguish the monad unit from the modification natural transformations α of Section 3.3.
- $\mu : T^2 \Rightarrow T$ is the **multiplication** natural transformation, with components $\mu_A : T(T(A)) \rightarrow T(A)$. It corresponds to the merging of sequential recognition cycles into a single net legal effect.

These data are subject to the monad laws:

$$\begin{aligned}\mu \circ T\eta_T &= \text{Id}_T && \text{(left unit law)} \\ \mu \circ \eta_T T &= \text{Id}_T && \text{(right unit law)} \\ \mu \circ T\mu &= \mu \circ \mu T && \text{(associativity law)}\end{aligned}$$

276

277 5.2 Legal Interpretation of the Monad Structure

278 Table 9 offers preliminary legal readings of the monad components. All entries are heuristic.

279 5.3 Significance and Limitations of the Monad Template

280 The monad construction at this stage is an analogy, not a theorem. Its value lies in the follow-
281 ing:

- 282 1. **Unified perspective:** The monad brings civil law’s two core moments — recognition of
283 facts and consolidation of effects — into a single structural framework.
- 284 2. **Revealing hidden assumptions:** The monad laws force us to ask whether legal recogni-
285 tion is idempotent (left unit law) and whether the sequencing of legal acts is associative
286 (associativity law). These are non-trivial legal questions that the monad template helps
287 to locate precisely.
- 288 3. **Connection to computation:** Monads are the standard abstraction for “computational
289 effects” in functional programming [3]. If the monad template can be made precise, it
290 would allow legal reasoning to be expressed as a monad transformer stack, with each

291 legal operation returning a value in the “civil law monad.”

292 The limitations are equally clear. The endofunctor T has not been explicitly constructed. The
 293 associativity law is legally problematic for ordered sequences of acts. The spaces of judicial
 294 discretion and value judgement in civil law are hard to encode in a deterministic functor. We
 295 regard these not as disqualifying the analogy, but as research questions to which the monad
 296 template gives a sharp formulation.

297 6 Introducing Time: From Static Structure to Dynamic Process

298 6.1 The Correct Temporal Model

299 The static partial category $\mathcal{R}$ cannot capture temporal factors such as statutes of limitations,
 300 age thresholds, or historical changes in legal status. An earlier and incorrect treatment of this
 301 problem proposed the *functor category* $[\mathcal{T}, \mathcal{R}]$ of functors $F : \mathcal{T} \rightarrow \mathcal{R}$. We now correct this.

302 A functor $F : \mathcal{T} \rightarrow \mathcal{R}$ assigns to each time instant t an *object* of $\mathcal{R}$, i.e. a single legal subject,
 303 and to each pair $t \leq t'$ a *morphism* of $\mathcal{R}$, i.e. a legal relation between two subjects. This is
 304 not what the temporal model intends: we want $F(t)$ to be the entire legal system at time t
 305 (a category, not a single object), and $F(t \leq t')$ to be the transition between two system states
 306 (a functor between categories, not a morphism between two subjects). The functor category
 307 $[\mathcal{T}, \mathcal{R}]$ therefore conflates two different categorical levels and is incorrect for this purpose.

Definition 6.1 — Temporal Category $\mathcal{T}$

Let $\mathcal{T} = (\mathbb{R}_{\geq 0}, \leq)$ be the poset of non-negative reals, viewed as a category:

- Objects: instants $t \in \mathbb{R}_{\geq 0}$
- Morphisms: a unique morphism $t \rightarrow t'$ exists if and only if $t \leq t'$
- Composition: given by transitivity of the order

308

Definition 6.2 — Temporalised Legal System

A *temporalised legal system* is a pseudofunctor

$$\Phi : \mathcal{T} \longrightarrow \mathbf{Cat}$$

where $\mathbf{Cat}$ is the (large) category of (small) categories. Concretely:

- For each $t \geq 0$, $\Phi(t) = \mathcal{R}_t$ is the partial category of civil legal relations valid at time t : it has the same class of objects (legal subjects alive at time t) but its morphism sets $\text{Hom}_{\mathcal{R}_t}(A, B)$ contain only those relations legally in force at t .
- For each $t \leq t'$, $\Phi(t \leq t') = \phi_{t,t'} : \mathcal{R}_t \dashrightarrow \mathcal{R}_{t'}$ is a partial functor expressing the “persistence and change” of relations from time t to time t' . It maps surviving relations to

309

their t' -versions and is undefined on relations that have been extinguished before t' .

- The pseudofunctor coherence conditions (“up to specified isomorphisms”) express the consistency requirement that composition of transitions $\phi_{t',t''} \circ \phi_{t,t'} \simeq \phi_{t,t''}$.

310

Remark 6.1. The pseudofunctor formulation replaces the incorrect $F : \mathcal{T} \rightarrow \mathcal{R}$. The key improvement is that $\Phi(t) = \mathcal{R}_t$ is a *category* (the whole legal system at time t), not a single object of some fixed $\mathcal{R}$. The transition $\phi_{t,t'}$ is a *functor between categories*, not a morphism in $\mathcal{R}$. This correctly handles the simultaneous creation of new relations, extinction of old ones, and modification of existing ones as time passes. The use of “pseudo” (rather than strict) functor acknowledges that the composition of transitions may only be isomorphic, rather than equal, to the direct transition — a standard feature of categorical models of time-dependent systems.

319 6.2 Modelling the Statute of Limitations

320 The statute-of-limitations system (Art. 188 Civil Code: ordinary limitation period of three
321 years) requires us to distinguish the substantive relation from its “actionability layer.”

322 **Start date.** Under Art. 188(2), the limitation period begins on the date *the right holder knows*
323 *or should know* (1) that their right has been infringed and (2) the identity of the obligor. This
324 is the correct general rule; the date of performance falling due is one specific scenario (where
325 the right to claim arises on that date) but is not the general starting point.

326 Within the temporalised system $\{\mathcal{R}_t\}$, a claim $r : A \rightarrow B$ has a two-layer structure:

- 327 • **Substantive layer** r : present in $\mathcal{R}_t$ for all t from creation until extinguished by perfor-
328 mance, set-off, or another substantive cause. The expiry of the limitation period does
329 *not* remove r from $\mathcal{R}_t$.
- 330 • **Actionability layer** $\tilde{r}$: present in $\mathcal{R}_t$ only while $t \leq t_0 + 3$ years, where t_0 is the date
331 the right holder knew or should have known of the infringement and the obligor. At
332 $t > t_0 + 3$ years, $\tilde{r}$ is removed from $\mathcal{R}_t$.
- 333 • **Defence right** $d : B \rightarrow A$: created (added to $\mathcal{R}_t$) when $\tilde{r}$ expires, conferring on B the
334 power to refuse performance.

335 7 Open Problems and Theoretical Limitations

336 Any work of formalisation must honestly acknowledge its limitations.

337 Of these problems, the non-formalisability of value judgements is the most fundamental. This

paper’s position is: the categorical framework can *locate* where value judgements occur in the structure (the “selection point” in a concurrent diagram, the “recognition entry” in the monad), but cannot *replace* those judgements. Formalisation makes the premises and consequences of judgements more transparent; it does not eliminate judgement.

8 Overview of Correspondences

9 Significance for Computational Law and Legal AI

9.1 Formal Foundation for Knowledge Representation

Current legal AI systems face the challenge of formally representing legal knowledge. Traditional rule-based systems struggle with the open texture of legal concepts [4], while machine-learning models face scrutiny regarding explainability and inferential transparency. The categorical framework offers a middle path:

- **Relational representation:** The morphism-centred structure of $\mathcal{R}$ is naturally suited to graph databases and knowledge graphs. Each morphism $r : A \rightarrow B$ of a given type (obligational, *in rem*, personal) can be encoded as a typed directed edge.
- **Legitimacy constraints:** The partial composition $\circ_{\text{Law}}$ corresponds to “valid inference paths” in automated reasoning — only composites recognised by law are permitted.
- **Natural transformations as update operations:** Modification (expressed as a natural transformation α) guides the dynamic update of a legal knowledge graph: any modification must preserve the coherence conditions expressed by the commutativity diagrams.

9.2 The Monad Template as a Computational Paradigm

Monads are the standard abstraction for computational effects in functional programming [3]. If the civil law monad template (T, η_T, μ) can be made precise, legal reasoning can be expressed as a monad-based computation:

- **Recognition effect:** η_T corresponds to a type-lifting operation `recognise: Fact -> LegalFact`.
- **Consolidation effect:** μ corresponds to combining a chain of legal acts into a net result, analogous to monadic `bind`.
- **Monad laws as correctness criteria:** A legal reasoning engine should satisfy the monad laws wherever they hold, ensuring consistency of inference across different orderings of recognition steps.

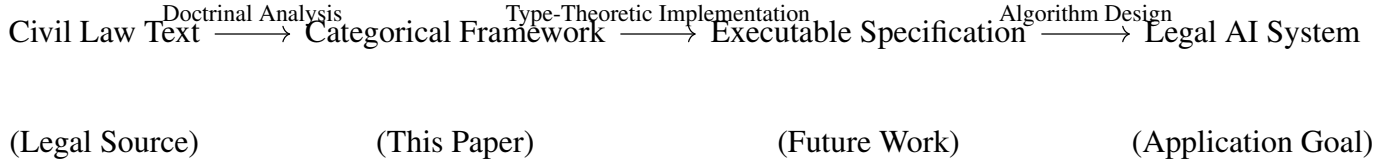


Figure 1: Transformation pathway from civil law text to a computable system

9.3 Limits/Colimits as Conflict Resolution Patterns

The two-stage decomposition of concurrence resolution — (i) identify the categorical structure of the conflict (the relevant limit or colimit diagram); (ii) apply the positive-law resolution rule (the “selection function”) — provides a clear architectural separation for legal reasoning systems:

- Stage (i) can be algorithmically implemented as a constraint graph analysis or unification problem.
- Stage (ii) encodes substantive legal rules as meta-rules applied after stage (i), making the legal basis for each resolution explicit and auditable.

9.4 The Temporal Pseudofunctor and Dynamic Reasoning

The pseudofunctor $\Phi : \mathcal{T} \rightarrow \mathbf{Cat}$ models temporal queries naturally:

- “Is claim $r : A \rightarrow B$ actionable at time t ?” corresponds to querying whether the actionability layer $\tilde{r}$ is in $\mathcal{R}_t$.
- “What rights did A hold against B at time t ?” corresponds to reading off $\text{Hom}_{\mathcal{R}_t}(A, B)$.
- The transition functors $\phi_{t,t'}$ record which relations survive from t to t' , supporting temporal legal tracing.

9.5 Pathway from Formalisation to Computabilisation

This paper occupies the second stage of Figure 1. Future research can build on this foundation to:

1. Implement $\mathcal{R}$ as a type family in a dependent type system [5];
2. Implement natural transformations and limit/colimit constructions as type-level computations;
3. Implement the monad template as an effect handler, once T is explicitly constructed;
4. Construct a prototype legal reasoning system with an explicit audit trail reflecting the two-stage concurrence architecture.

393 10 Conclusion

394 This paper has attempted to establish a categorical formal framework for civil legal relations,
 395 and in doing so has corrected several technical errors present in earlier formulations. We con-
 396 struct $\mathcal{R}$ as a *partial* category (not a full category) from the outset, with lawful composition
 397 $\circ_{\text{Law}}$ defined only where law recognises a derived relation. The three core operations — cre-
 398 ation, extinction, and modification — are distinguished by their categorical character: creation
 399 is functorial (extension); extinction is non-functorial (discontinuous removal, with cascading
 400 effects); modification is expressed by natural transformations α (with notation kept strictly
 401 separate from the monad unit η_T).

402 For concurrence, we correct the mis-identification of the coequalizer with proportional dis-
 403 tribution: the coequalizer is a quotient construction that identifies two morphisms' effects,
 404 not a model of pro-rata allocation. Limits and colimits supply the structural form of conflict;
 405 positive law supplies the resolution.

406 For temporality, we correct the use of the functor category $[\mathcal{T}, \mathcal{R}]$, which conflated categorical
 407 levels: under that formulation, $F(t)$ would be a legal subject, not the entire relational system
 408 at time t . The correct formulation is a pseudofunctor $\Phi : \mathcal{T} \rightarrow \mathbf{Cat}$ under which $\Phi(t) = \mathcal{R}_t$
 409 is a category. The statute of limitations is modelled via a two-layer distinction (substantive
 410 relation vs. actionability layer), with the correct legal start date (when the right holder knew or
 411 should have known).

412 For the monad, we replace the ungrounded claim “ $T = F \circ U$ ” with an honest statement that
 413 T is programmatic, and we flag the associativity law as the weakest point of the analogy,
 414 suggesting downgrade to a lax monad if full associativity cannot be established.

415 The main contributions are:

- 416 1. **Structural clarification:** The partial-category framework faithfully reflects civil law's
 417 conditional composition of legal effects.
- 418 2. **Correct categorical vocabulary:** Equalizers, coequalizers, limits, and colimits are used
 419 with their correct mathematical meanings, with legal interpretations that do not over-
 420 reach those meanings.
- 421 3. **Corrected temporal model:** The pseudofunctor $\Phi : \mathcal{T} \rightarrow \mathbf{Cat}$ correctly assigns a cate-
 422 gory (not an object) to each time instant.
- 423 4. **Honest monad template:** The monad analogy is presented as a structural scaffold with
 424 clearly identified open problems (construction of T , validity of associativity), not as a
 425 completed model.

5. **Computation connection:** The two-stage concurrence architecture and the monad-based computation paradigm provide actionable theoretical guidance for legal AI system design.

The work remains preliminary. A rigorous development would require an explicit construction of T , a proof (or refutation) of the monad laws for specific families of civil law operations, and an implementation of the partial-category structure in a proof assistant. We offer this paper as a conceptual starting point for that programme.

References

- [1] .. 5., 2019.
- [2] .., 2020.
- [3] Eugenio Moggi. “Notions of computation and monads”. In: *Information and Computation* 93.1 (1991), pp. 55–92.
- [4] H. L. A. Hart. “Theory and Practice in Jurisprudence”. In: *Proceedings of the Aristotelian Society* (1955).
- [5] Michael Theunissen and Lisa Goosen. “A sufficient condition for the existence of initial algebras in categories of functors”. In: *Journal of Logic and Algebraic Programming* (2014).

Table 9: Monad components and their legal interpretations (heuristic analogies only)

Component	Mathematical Meaning	Legal Interpretation
T (endofunctor)	Maps objects and morphisms to their “normalised” versions	Civil law’s normative filtering: raw relational data is evaluated, classified, and assigned legal force. T is programmatic; its explicit construction requires further work.
η_T (unit)	$\text{Id}_{\mathcal{R}} \Rightarrow T$; lifts every object/morphism into T ’s image	Legal recognition: a social fact (represented by Id) is “lifted” into the space of legal relations by being formally recognised as a legal fact. Corresponds to the entry conditions for legal evaluation.
μ (multiplication)	$T^2 \Rightarrow T$; collapses double application of T into one	Merging of sequential legal effects: two rounds of recognition (e.g. a chain of juridical acts, each generating a new layer of effects) are consolidated into a single net effect.
Left unit law	$\mu \circ T\eta_T = \text{Id}_T$	A relation that has already been recognised once does not acquire a different normative status by being recognised again. Structurally analogous to the principle of <i>res judicata</i> or the idempotence of legal recognition.
Right unit law	$\mu \circ \eta_T T = \text{Id}_T$	Embedding an already-recognised relation into a second round of recognition and then collapsing yields the original. Corresponds to the idea that a validly formed legal act does not need re-validation to remain operative.
Associativity law	$\mu \circ T\mu = \mu \circ \mu T$	The order in which a sequence of legal acts is “bracketed” for consolidation does not affect the net result. This is the most contentious of the three laws: the order of juridical acts is often legally significant (e.g. offer must precede acceptance). The law can at best be claimed for specific subclasses of legal sequences, such as parallel independent defences whose cumulative effect is order-invariant. ²

Table 10: Temporal factors and their modelling in the pseudofunctor framework

Temporal Factor	Legal Mechanism	Modelling in $\{\mathcal{R}_t\}$
Statute of limitations ()	Actionability of a claim is diminished after 3 years from the date the right holder knew or should have known	Substantive r persists; actionability layer $\tilde{r}$ is removed from $\mathcal{R}_t$ at $t = t_0 + 3$ years; defence right d created
Peremptory period ()	The right itself is extinguished upon expiry of a statutory period	Relation r itself is removed from $\mathcal{R}_t$ at expiry, with cascading extinction of dependents
Acquisitive prescription ()	A right is acquired through possession for a certain period	A new morphism r is added to $\mathcal{R}_t$ at the end of the prescription period
Age threshold	Civil capacity changes with age (Art. 17–19 Civil Code)	The “attribute” of subject A undergoes a step-change at $t = 18$ years from A ’s birth; this affects which morphisms in $\text{Hom}_{\mathcal{R}_t}(A, -)$ are legally valid

Table 11: Major limitations of the framework and possible responses

Problem	Nature	Possible Response
Non-existence of limits/colimits	Diagrams in $\mathcal{R}$ need not have limits or colimits, yet law must always resolve concurrences	The <i>judicial completeness operator</i> J models the court's power to impose a resolution as an external supplement to the categorical structure; it is not itself a categorical construction
Partial definedness of composition	Lawful composition $\circ_{\text{Law}}$ is not everywhere defined	The partial-category framework is the correct setting; full category theory should not be assumed
Non-formalisation of value judgements	Principles such as good faith and proportionality cannot be reduced to the determinism of a functor	Acknowledge this as the principal boundary of the formal framework; use nondeterminism (e.g. a relation-valued “functor”) to model discretion ranges, while recognising that substantive choices remain outside the model
Associativity of juridical acts	The monad's associativity law may not hold for ordered sequences of acts	Downgrade to a lax monad or semi-monad if full associativity cannot be established; or restrict the monad claim to specific act-families
Bilateral obligations	Directed morphisms $A \rightarrow B$ capture only one side of a bilateral obligation	Use pairs $(r_1 : A \rightarrow B, r_2 : B \rightarrow A)$ or an enriched categorical setting that encodes bilateral structure directly
Multiple legal interpretations	The same social fact may receive different legal classifications under different interpretive frameworks	Express different frameworks as different functors F, G ; natural transformations express transitions between frameworks
Openness of the legal system	Civil law absorbs social facts and evolves continuously	The pseudofunctor $\Phi : \mathcal{T} \rightarrow \mathbf{Cat}$ already accommodates this; the evolution of law itself (legislative change) requires a further meta-level functor

Table 12: Summary of correspondences between civil law concepts and categorical constructions

Civil Law Concept	Categorical Construction	Key Properties / Remarks
Legal subject	Object of $\mathcal{R}$	Defined through relational networks it participates in
Civil legal relation	Morphism in $\text{Hom}(A, B)$	Directed (convention); bilateral structure handled by pairs
Creation	Extension $\iota : \mathcal{R} \hookrightarrow \mathcal{R}'$	Adds morphisms (and sometimes objects); functorial
Extinction	Removal of morphism r ; cascade to composites through r	Discontinuous; no natural functor from $\mathcal{R}$ to $\mathcal{R}'$
Modification (content)	$\alpha : \text{Id} \Rightarrow \text{Id}$; commutativity condition in partial sense	Does not change subjects; adjusts obligation content
Modification (subject)	$\alpha : \text{Id} \Rightarrow G$; G is subject-replacement functor	Same obligation content, new holder; models assignment correctly
Concurrence of rights	Convergent diagram in $\mathcal{R} + \text{limit/colimit where they exist}$	Categorical structure gives form; positive law gives resolution
Temporal system	Pseudofunctor $\Phi : \mathcal{T} \rightarrow \mathbf{Cat}$; $\Phi(t) = \mathcal{R}_t$	Replaces incorrect functor-category formulation
Legal recognition	Monad unit $\eta_T : \text{Id} \Rightarrow T$	Lifts social facts into legal evaluation space
Consolidation of effects	Monad multiplication $\mu : T^2 \Rightarrow T$	Merges sequential recognition cycles into net effect
Partial lawful composition	Partial category structure on $\mathcal{R}$	Composition defined only when law recognises derived relation
Judicial resolution	Completeness operator J (external to $\mathcal{R}$)	Supplements categorical incompleteness of $\mathcal{R}$